基于核函数的软件定义网络 DDoS 实时安全系统 *

刘　敏，滕　华，何先波

(西华师范大学 计算机学院，四川 南充 637009)

摘　要：针对软件定义网络中 DDoS 攻击的检测准确率与延迟较长的问题，提出了一种基于核函数的软件定义网络 DDoS 实时安全系统。首先，每个周期提取软件定义网络的报文头信息，并组织成矩阵形式；其次，采用马氏距离分析相邻特征向量的显著变化，设计了两个核函数综合评估攻击行为的流量；最终，采用谱聚类技术与协方差统计信息自动地定位攻击者。基于真实软件定义网络进行了实验，结果显示该安全系统实现了较高的检测准确率，并且实现了理想的处理时间。

关键词：软件定义网络；网络安全；拒绝服务攻击；核函数；谱聚类技术

中图分类号：TP393　　doi: 10.19734/j.issn.1001-3695.2018.09.0659

Real time DDoS security system of software definition networks based on kernel functions

Liu Min, Teng Hua, He Xianbo

(*Computer Academy, China West Normal University, Nanchong Sichuan* 637009, *China*)

Abstract: Aiming at the problems of log detection accuracy and long delay of DDoS(Distributed Denial-of-Service) attacks in the software definition networks, This paper proposed a real time DDoS security system of software definition networks based on kernel functions. Firstly, it abstracted the packet header fields of software definition networks periodly, and formed the abstracted information as matrices; then, it adopted the Mahalanobis distance to analyze the significant change of continuous feature vectors, and it designed two kernel functions to evaluate the behavior flows of attacks; lastly, the attackers are identified by the spectral clustering technique and the covariance statistical information. Experimental results based on the real software definition networks show that the proposed security system realizes a good detection accuracy, and performs a reasonable processing time.

Key words: software definition network; network security; denial-of-service attack; kernel function; spectral clustering

0　引言

DDoS(distributed denial of service)攻击是当前互联网中最为常见的攻击行为[1]。DDoS 的实现简单，并且实现的成本低，因此在各种类型的网络中 DDoS 均为一个重要的威胁[2]。软件定义网络(software defined networking, SDN)实现了网络控制与数据分离的思想，支持高度的开放性与可编程性，但是安全问题限制了 SDN 在诸多场景下的大规模部署与应用。SDN 的安全机制主要可分为六种类型[3]：a)SDN 安全控制器；b)可组合的安全模块库；c)控制器的 DoS、DDoS 攻击防御系统；d)流规则的合法性与一致性检测；e)北向接口的安全性；f)应用程序的安全性。其中 DDoS 攻击直接导致网络瘫痪，对 SDN 的危害极大[4]。

文献[5]利用 GHSOM 技术，设计了基于对象特征的 DDoS 攻击检测方法。该方法结合 SDN 网络及攻击特点，提出基于目的地址的检测七元组，并以此作为判断目标地址是否受到 DDoS 攻击的检测元素。文献[6]提出了一种在 SDN 环境下基于 KNN 算法的模块化 DDoS 攻击检测方法，该方法选取 SDN 网络的五个关键流量特征，采用优化的 KNN 算法对选取的流量特征进行流量异常检测。文献[7]提出了一种控制器的 DDoS 检测系统，该系统首先将端口的流事件分类，使用序贯概率比检验(sequential probability ratio test, SPRT)检测流量是否为异常。文献[8]对几类 SDN 的 DDoS 安全系统进行了分析，基于流量的统计分析方法一般检测效果较好，但是计算效率较低；而基于 SDN 报文头特征的分析方法计算效率较高，但是检测的准确率较低。

DDoS 攻击可能导致整个网络瘫痪，所以需要在初期就能识别出 DDoS 流量，以防止 DDoS 造成更大的破坏[9]。为了在保证 DDoS 安全系统检测准确率的前提下保持较快的处理速度，设计了一种基于核函数的软件定义网络 DDoS 实时安全系统。DDoS 攻击的出现往往伴随着流量模式的剧烈变化，所以本系统将消息流特征的明显变化作为一个潜在的 DDoS 攻击。目前主流的 DDoS 安全系统主要检测出 DDoS 攻击的流量，无法定位攻击源，而本系统能够检测出 DDoS 攻击行为，并且识别出攻击者。本系统属于无监督方法，利用观察的消息流量类型与数据量，并不需要额外的信息。

1　问题模型

1.1　软件定义网络的 OpenFlow 模型

控制器管理所有的路由决策，通过 OpenFlow 交换机的 flow 表完成每个报文的转发。图 1 所示是 OpenFlow version 1.3 协议[10]的报文头。

Ingress port	Ether src	Ether dest	VLAN ID	VLAN priority	IP_SRC	IP_dest	IP protocal	Src port	Dest port

图 1　OpenFlow version 1.3 协议的报文头

Fig. 3　Message header of OpenFlow version 1.3 protocal

收稿日期：2018-09-05；**修回日期：**2018-10-25　**基金项目：**四川省科技厅项目（2018GFW0151）；西华师范大学校级教改项目（xjjgxh2017134）

作者简介：刘敏（1977-），女，重庆人，副教授，硕士，主要研究方向为物联网(liangj_sheng@126.com)；滕华（1972-），男，四川广安人，副教授，硕士，主要研究方向为软件工程；何先波（1971-），男，四川苍溪人，教授，博士，主要研究方向为物联网.

图 2 所示是 OpenFlow 报文的处理流程。

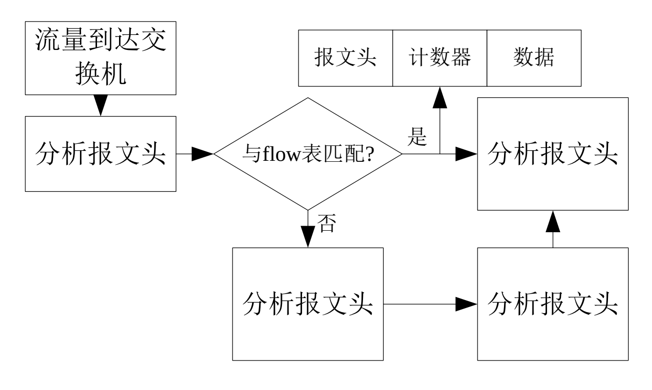


图 2　OpenFlow 报文的处理流程

Fig. 2　Processing flow of openflow message

1.2　软件定义网络的 DDoS 攻击模型

每隔一个周期对控制报文进行一次采样，表示为 $t=i\Delta t$，将该时段的样本组成一个特征向量。Δt 表示一个观察时段，监控该时段交换机收到的控制报文。在 Δt 的结束，将 IP_SRC 用户的流量表示为 d 维的向量 $\mathbf{v}_r$，其中 d 为各种流量类型的数量。向量 $\mathbf{v}$ 的元素为整型，元素对应了第 i 时间帧 $((i-1)\Delta<t<i\Delta)$内每种消息类型的出现次数。

假设第 r 个用户在观察时段内发出 P_r 个消息，每个消息的计数器表示为 $\mathbf{v}_r^p$, $p=1,\ldots,P_r$，因此观察时段内的总采样信息可表示为 $\mathbf{v}_r=\sum_{p=1}^{P_r}\mathbf{v}_r^p$，$\mathbf{v}_r$ 表示第 r 个用户发送的计数器矩阵，如图 3 所示。

0	1	…	0	…	0	$\mathbf{v}_{1,r}$
0	1	…	0	…	0	$\mathbf{v}_{2,r}$
0	1	…	0	…	0	$\mathbf{v}_{3,r}$
…	…	…	…	…	…	…
0	1	…	0	…	0	$\mathbf{v}_{d-1,r}$
0	1	…	0	…	0	$\mathbf{v}_{d,r}$
$\mathbf{v}_r^1$	$\mathbf{v}_r^2$		$\mathbf{v}_r^p$		$\mathbf{v}_r^{P_r}$	$\mathbf{v}_r$

图 3　第 r 个用户的计数器矩阵

Fig. 3　Counter matrix of rth user

假设 $\mathbf{x}$ 为状态向量，表示一个 d 维的计数向量，定义为在一个观察时段内$|U|$个用户的协同活动次数。服务器端所有用户的计数器向量之和定义为服务器状态向量，计算式为 $\mathbf{x}=\sum_{r=1}^{|U|}\mathbf{v}_r$，如图 4 所示。

$v_{1,1}$	$v_{1,2}$	…	$v_{1,r}$	…	$v_{1,\|U\|}$	$\mathbf{x}_1$
$v_{2,1}$	$v_{2,2}$	…	$v_{2,r}$	…	$v_{2,\|U\|}$	$\mathbf{x}_2$
$v_{3,1}$	$v_{3,2}$	…	$v_{3,r}$	…	$v_{3,\|U\|}$	$\mathbf{x}_3$
…	…	…	…	…	…	…
$v_{d-1,1}$	$v_{d-1,2}$	…	$v_{d-1,r}$	…	$v_{d-1,\|U\|}$	$\mathbf{x}_{d-1}$
$v_{d,1}$	$v_{d,2}$	…	$v_{d,r}$	…	$v_{d,\|U\|}$	$\mathbf{x}_d$
$\mathbf{v}_1$	$\mathbf{v}_2$		$\mathbf{v}_r$		$\mathbf{v}_{\|U\|}$	$\mathbf{X}$

图 4　服务器状态向量的示意图

Fig. 4　Diagram of server states vector

假设 $\mathbf{x}_i$ 与 $\mathbf{x}_j\in R^d$ 分别表示第 i、j 个观察时段的服务器状态向量，本文采用这些特征向量(服务器状态向量)监控软件定义网络的流量变化。假设 $\mathbf{M}$ 为一个 $d\times d$ 的正定矩阵，$D_{\mathbf{M}}(x_i,x_j)$为特征向量 $\mathbf{x}_i$ 与 $\mathbf{x}_j$ 之间的距离，$\mathbf{M}$ 表示尺度矩阵。$f(\mathbf{M}|\mathbf{x}_n:\mathbf{x}_{n-k-1})$为一个关于 $\mathbf{M}$ 的函数，定义为 $\mathbf{x}_{n-k-1}$ 到 $\mathbf{x}_n$ 之间(时间长度为 k)窗口的特征向量记录。

上述特征向量不包含时间戳信息，假设第 r 个用户、消息 P_r 的时间戳序列为 $t_r^1,\ldots,t_r^{P_r}$，通过对消息指示向量增加增广向量引入时间戳信息，具体方法为：$(\mathrm{w}_r^p)^{\mathrm{T}}=((\mathrm{v}_r^p)^{\mathrm{T}},t_r^p)$。$\mathrm{w}_r^p$ 表示带时间戳的消息指示向量，$\mathrm{w}_r^p\in\mathrm{R}^{d+1}$ 与向量 $\mathbf{v}_r^p$ 的关系如图 5 所示。

$$\mathbf{w}_r^p=\begin{vmatrix}\mathbf{v}_r^p\\ t_r^p\end{vmatrix}$$

图 5　增加时间戳的用户消息向量

Fig. 5　Time-stamped user message vector diagram of serv

可将任意用户 u_r 表征为一个时间的序列，定义为一个矩阵：$\mathbf{V}_r=[\mathbf{v}_r^1|\mathbf{v}_r^2|\ldots|\mathbf{v}_r^{P_r}]$ 或者 $\mathbf{W}_r=[\mathbf{w}_r^1|\mathbf{w}_r^2|\ldots|\mathbf{w}_r^{P_r}]$。采用核函数计算两个用户($u_q$, u_r)的相似性，将相似性表示为 $K(u_q, u_r)$。$\kappa(\mathbf{w}_r^{Pr},\mathbf{w}_r^{Pq})$ 定义为相同时段内两个用户之间的相似性，其中 $\mathbf{w}_r^{Pr}$ 表示第 r 个用户的第 p_r^{th} 个消息，$\mathbf{w}_r^{Pq}$ 表示第 q 个用户的第 p_q^{th} 个消息。最终可计算出某个观察时段内$|U|\times|U|$(所有用户)的核矩阵 $\mathbf{K}$。

2　基于自适应距离的异常监控

在一个平稳过程中消息的特征应当具有高度的统计相似性，而非稳态过程的两个特征集之间的距离较大。因此检测消息特征距离的显著变化点，对采样时间的流量进行分析，检测出潜在的 DDoS 攻击行为。

2.1　马氏距离

假设 $\mathbf{M}\in\mathbf{S}_+$为一个 $d\times d$ 的对称半正定矩阵，那么 $\mathbf{x}_i$ 与 $\mathbf{x}_j$ 之间的马氏距离 D_M 计算为

$$D_M(\mathbf{x}_i, \mathbf{x}_j)=(\mathbf{x}_i-\mathbf{x}_j)^{\mathrm{T}}\mathbf{M}(\mathbf{x}_i-\mathbf{x}_j) \tag{1}$$

全秩样本协方差矩阵的逆 $\mathbf{\Sigma}$ 是马氏距离的一种特殊情况，如果特征集服从标准的高斯分布，可得 $\mathbf{M}=\mathbf{\Sigma}=\mathbf{I}$。对称半正定矩阵可分解为 $\mathbf{M}=\mathbf{A}^{\mathrm{T}}\mathbf{A}$，因此 $\mathbf{A}$ 为一个 $e\times d$ 的投影矩阵，并且 $e\leqslant d$。可获得以下的关系式：

$$\begin{aligned}D_{\mathrm{M}}(\mathbf{x}_i,\mathbf{x}_j)&=(\mathbf{x}_i,\mathbf{x}_j)^{\mathrm{T}}\mathbf{M}(\mathbf{x}_i,\mathbf{x}_j)\\&=(\mathbf{x}_i,\mathbf{x}_j)^{\mathrm{T}}\mathbf{A}^{\mathrm{T}}\mathbf{A}(\mathbf{x}_i,\mathbf{x}_j)=\left(\mathbf{A}(\mathbf{x}_i,\mathbf{x}_j)\right)^{\mathrm{T}}\mathbf{A}(\mathbf{x}_i-\mathbf{x}_j)\\&=\|\mathbf{a}_i-\mathbf{a}_j\|_2^2=D_E(\mathbf{a}_i,\mathbf{a}_j)=D_{\mathrm{A}}(\mathbf{x}_i,\mathbf{x}_j)\end{aligned} \tag{2}$$

其中：$\mathbf{a}_i=\mathbf{A}\mathbf{x}_i$ 为投影向量；$\mathbf{D}_E$ 为欧氏距离。式(2)说明特征空间中马氏距离等价于投影空间的欧氏距离。

2.2　基于距离的网络流量模型

通过观察滑动窗口的距离之和实现对网络流量的监控，该距离称为流量的移动距离，将滑动窗口之和与阈值 ϵ 比较，决定是否为异常流量。一个大小为 k 的窗口移动距离可定义为一个关于对称正定矩阵($\mathbf{M}\in\mathbf{S}_{++}$)的函数，如式（3）所示。

$$f(\mathbf{M}|\mathbf{x}_n:\mathbf{x}_{n-k-1})=\sum_{j=n-k-1}^{n-1}(\mathbf{x}_n-\mathbf{x}_j)^{\mathrm{T}}\mathbf{M}(\mathbf{x}_n-\mathbf{x}_j) \tag{3}$$

如果使用马氏距离计算的移动距离高于阈值 ϵ，即 $f(\mathbf{M}_{n-1}|\mathbf{x}_n:\mathbf{x}_{n-k-1})>\epsilon_{th}$，那么触发一个警告。每个周期更新一次马氏距离，定义为

$$\min_{\mathbf{M}\in\mathbf{S}_{++}} f(\mathbf{M}|\mathbf{x}_n:\mathbf{x}_{n-k-1})+\lambda D_{ld}(\mathbf{M},\mathbf{M}_{n-1})+\beta D_{ld}(\mathbf{M},\mathbf{I}) \tag{4}$$

其中：第 2、3 项为正则化函数，该函数基于对数行列式散数实现(LogDet)[11]。LogDet 函数可估计两个矩阵之间的距离，定义为：

$$D_{ld}(\mathbf{M},\mathbf{M}_{t-1}) = tr(\mathbf{M}\mathbf{M}_{t-1}^{-1}) - \log\det(\mathbf{M}\mathbf{M}_{t-1}^{-1}) - d \tag{5}$$

其中：$tr(.)$为矩阵的迹函数。

计算式(4)的导数，可获得最优的马氏矩阵($\mathbf{M}^*$)：

$$\mathbf{M}^* = \left(\frac{\lambda}{\lambda+\beta}\mathbf{M}_{n-1}^{-1} + \frac{\beta}{\lambda+\beta}\mathbf{I} + \frac{1}{\lambda+\beta}\sum_{j=n-k-1}^{n-1}(\mathbf{x}_n-\mathbf{x}_j)(\mathbf{x}_n-\mathbf{x}_j)^{\mathrm{T}} \right)^{-1} \tag{6}$$

每个周期重复更新该马氏矩阵。算法 1 所示是流量改变的检测算法。

算法 1　自适应移动距离的流量改变检测算法

```
初始化 M0;
初始化参数 k,λ,β,α;
while (新流量) {
    观察窗口(大小为 k)内的流量，计算计数器向量;
    if f(Mn-1|xn:xn-k-1) > ϵ {
        发出警告;
        允许恶意用户检查程序;
    }
    评估 M*;
    Mn-1=M*;
}
```

2.3　移动距离的阈值

根据实验分析，移动距离之和的分布可近似为卡方分布。然后从高斯分布获得马氏距离，因此可得：$\mu=\mathbf{x}_n$，$\boldsymbol{\Sigma}=\mathbf{M}^{-1}$。假设 $\mathbf{y}$ 是当前滑动窗口的观察集合，如果 $\mathbf{y}$ 是一个服从高斯分布的 d 维随机向量，其平均向量为 μ，协方差矩阵为 $\boldsymbol{\Sigma}$，将 $z=(\mathbf{y}-\mathbf{x}_n)^{\mathrm{T}}\mathbf{M}(\mathbf{y}-\mathbf{x}_n)=(\mathbf{y}-\mu)^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{y}-\mu)$转换为自由度为 d 的卡方分布。

假设 z_i 表示 k 个独立同分布的随机变量，z_i 服从卡方分布，如 $z_1 \sim \chi^2_{\alpha,d_1}, z_2 \sim \chi^2_{\alpha,d_2}, \ldots, z_k \sim \chi^2_{\alpha,d_k}$。根据独立卡方分布的变量属性，随机变量之和服从卡方分布，其自由度为 $d_1+d_2+\ldots+d_k$。具体可表示为

$$Z = z_1 + z_2 + \ldots + z_k,\ \ Z \sim \chi^2_{\alpha,d_1+d_2+\ldots+d_k} \tag{7}$$

最终，异常流量检测模型的阈值为 $\varepsilon_{th} = \chi^2_{\alpha,k*d}$，其中 α 表示收到异常流量的概率。将正常流量情况下，移动距离平均值的评分表示为 Z，Z 值达到阈值 ε_{th} 的概率为 α，α 值根据应用场景可设为 $\alpha\epsilon\{0.1,0.05,0.02,0.01\}$。

异常流量检测模型的阈值计算式为

$$\varepsilon_{th} = ck\left(\frac{d}{2}\right)^2 \tag{8}$$

其中：k 为时间窗口长度；d 为向量维度；c 为一个常量。因为每个周期地观察中均会更新观察量之间的马氏距离，所以本系统是一个自适应的智能安全系统。

3　判断恶意用户

如果一个异常流量为一个 DDoS 攻击，那么需要识别恶意用户的集合，从而防止造成分布式攻击。观察周期内每个用户的历史行为表示为一个时间序列，如图 3 所示。使用相似性函数处理时间序列，将行为相似的用户分类为同一个组。提出了两种攻击判别方法：a)基于全局的时间序列对齐核，计算消息序列特征之间的距离；b)在观察周期结尾提取的用户消息数量向量。

3.1　时间序列对齐核

本文考虑 k 长度窗口内的带时间戳消息，将消息表示为时间序列格式，即$(n-k-1),\ldots,(n-1)$。每个用户的序列包含不同数量的消息事件，每个事件发生的时间点不同。本文的目标是基于核方法计算用户消息之间的相似性，首先需要将消息队列进行对齐处理，不同长度消息之间的相似性较低，因此同类型消息的相似性较高，不同类型消息的相似性较低。

假设一对用户(u_q, u_r)的带时间戳消息序列为$(\mathbf{W}_q，\mathbf{W}_r)$，假设 $\mathbf{W}_q=[\mathbf{w}_q^1|\mathbf{w}_q^2|\mathbf{w}_q^3]$ 与 $\mathbf{W}_r=[\mathbf{w}_r^1|\mathbf{w}_r^2]$ 分别具有 3 个与 2 个消息事件。图 6 所示是 $\mathbf{W}_q$ 与 $\mathbf{W}_r$ 所有可能的对齐情况，该案例共有 5 个对齐情况，如下所示：

① $(\mathbf{w}_q^1,\mathbf{w}_r^1),(\mathbf{w}_q^1,\mathbf{w}_r^2),(\mathbf{w}_q^2,\mathbf{w}_r^2),(\mathbf{w}_q^3,\mathbf{w}_r^2)$

② $(\mathbf{w}_q^1,\mathbf{w}_r^1),(\mathbf{w}_q^2,\mathbf{w}_r^2),(\mathbf{w}_q^3,\mathbf{w}_r^2)$

③ $(\mathbf{w}_q^1,\mathbf{w}_r^1),(\mathbf{w}_q^2,\mathbf{w}_r^1),(\mathbf{w}_q^2,\mathbf{w}_r^2),(\mathbf{w}_q^3,\mathbf{w}_r^2)$

④ $(\mathbf{w}_q^1,\mathbf{w}_r^1),(\mathbf{w}_q^2,\mathbf{w}_r^1),(\mathbf{w}_q^3,\mathbf{w}_r^2)$

⑤ $(\mathbf{w}_q^1,\mathbf{w}_r^1),(\mathbf{w}_q^2,\mathbf{w}_r^1),(\mathbf{w}_q^3,\mathbf{w}_r^1),(\mathbf{w}_q^3,\mathbf{w}_r^2)$

文献[12]提出了一种全局对齐核，使用动态规划计算两个序列的相似性，本文基于文献[12]的算法实现，唯一的修改是引入马氏距离。

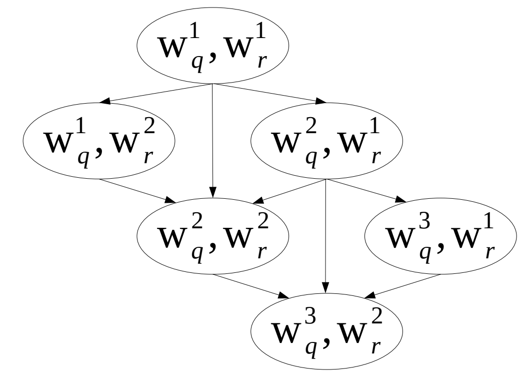


图 6　$\mathbf{W}_q$ 与 $\mathbf{W}_r$ 的所有对齐情况

Fig. 6　All aligned cases of $\mathbf{W}_q$ and $\mathbf{W}_r$

3.1.1 全局序列对齐核

给定两个用户(u_q, u_r)的两个消息序列 $\mathbf{W}_q=[\mathbf{w}_q^1|\mathbf{w}_q^1|\ldots|\mathbf{w}_q^{P_q}]$ 与 $\mathbf{W}_r=[\mathbf{w}_r^1|\mathbf{w}_r^1|\ldots|\mathbf{w}_r^{P_r}]$，假设其状态空间为 Ω，设计一个二维数组 $T_{Pq,Pr}$ 保存二维序列，其中 $T_{Pq,0}=0(p_q=1,\ldots,P_q)$，$T_{0,Pr}=0(p_r=1,\ldots,P_r)$，$T_{0,0}=0$。假设存在一个函数可度量用户 u_q 与 u_r 之间消息事件的相似性，设为 $\kappa(\mathbf{w}_q^{p_q},\mathbf{w}_r^{p_r})$。因此可通过递归方法计算出 $T_{Pq,Pr}$：

$$T_{p_q,p_r} = \left(T_{p_q,p_r-1} + T_{p_q-1,p_r-1} + T_{p_q-1,p_r}\right)\kappa(\mathbf{w}_q^{p_q},\mathbf{w}_r^{p_r}) \tag{9}$$

最终可获得两个用户(u_q, u_r)之间相似性未正则化的结果，如式(10)所示。

$$K_{unnormed}(u_q,u_r) = T_{P_q,P_r} \tag{10}$$

获得每对用户的核矩阵之后，为了解决尺度不统一的问题，对$|U|\times|U|$核矩阵进行单位对角正则化处理，其中$|U|$表示系统中活动用户的数量。单位对角正则化的方法如式（11）所示。

$$K(u_q,u_r)=\frac{K_{unnormed}(u_q,u_r)}{\sqrt{K_{unnormed}(u_q,u_q)}\sqrt{K_{unnormed}(u_r,u_r)}}\cdot q$$

$$K(u_q,u_r)\to[0,1] \tag{11}$$

将上述核矩阵称为时间序列核。

3.1.2 核函数

一个窗口内的每个用户可表征为带时间戳的消息序列，用户消息序列包括了不同的长度与不同的消息类型。

假设两个带时间戳的向量为 $(\mathbf{w}_q^{p_q})^{\mathrm{T}}=((\mathbf{v}_q^{p_q})^{\mathrm{T}},t_q^{p_q})$ 与 $(\mathbf{w}_r^{p_r})^{\mathrm{T}}=((\mathbf{v}_r^{p_r})^{\mathrm{T}},t_r^{p_r})$，两个向量的核函数定义为

$$\kappa(\mathbf{w}_q^{p_q},\mathbf{w}_r^{p_r})=\exp(-\gamma D_M(\mathbf{v}_q^{p_q},\mathbf{v}_r^{p_r})-\rho|t_q^{p_q}-t_r^{p_r}|)$$

$$D_M(\mathbf{v}_q^{p_q},\mathbf{v}_r^{p_r})=(\mathbf{v}_q^{p_q},\mathbf{v}_r^{p_r})^{\mathrm{T}}\mathbf{M}(\mathbf{v}_q^{p_q},\mathbf{v}_r^{p_r}) \tag{12}$$

其中：$\mathbf{M}$ 为式(6)的马氏矩阵。如果 $\mathbf{v}_q^{p_q}=\mathbf{v}_r^{p_r}$ 并且 $t_q^{p_q}=t_r^{p_r}$，可得 $\kappa(\mathbf{w}_q^{p_q},\mathbf{w}_r^{p_r})=1$。系数 γ 与 ρ 分别为消息距离与时间距离的权重，本文假设两者相等 $\gamma=\rho=1$。

3.2　用户距离核

基于马氏距离计算用户之间的相似性核矩阵，如果马氏距离越接近 0，那么两个用户的相似性越高，反之则相似性越低，马氏距离核可视为一种高斯核的特殊情况。基于用户的消息计数向量 $\mathbf{v}_q,\mathbf{v}_r\in\mathrm{R}^d$，比较两个用户 u_q 与 u_r 的相似性：

$$K(u_q,u_r)=\exp(-(\mathbf{v}_q-\mathbf{v}_r)^{\mathrm{T}}\mathbf{M}(\mathbf{v}_q-\mathbf{v}_r)) \tag{13}$$

将式(13)的核简称为距离核，如果 $\mathbf{v}_q=\mathbf{v}_r$，那么 $K(u_q,u_r)=1$，该特征向量并未包含消息的时间戳信息，仅仅评价了该窗口内用户的消息特征。

3.3　谱聚类算法

式(11)(13)两式中计算了用户之间相似性的核矩阵 $\mathbf{K}$，矩阵 $\mathbf{K}$ 可表示为一个全连接的加权邻接图，图中顶点表示用户，边表示相似性。邻接矩阵应当可分为恶意用户与合法用户两个子图。使用拉普拉斯谱聚类算法将邻接图分类，该方法可将相似的节点分为同一类，同时保证不相似节点彼此远离[13]。

将第 q 个活动用户核矩阵的元素之和定义为该用户的度，那么一个给定窗口中第 q 个活动用户的度定义为

$$dg_q=\sum_{r=1}^{|U|}\mathbf{K}_{q,r} \tag{14}$$

其中：$\mathbf{K}_{q,r}=\mathbf{K}(u_q,u_r)$。

度矩阵 $\mathbf{D}$ 为对角矩阵，其对角元素为“度”：dg_1，$dg_2,\ldots,dg_{|U|}$。拉普拉斯矩阵 $\mathbf{L}$ 的评估方法如式（15）定义：

$$\mathbf{L}=\mathbf{D}\text{-}\mathbf{K} \tag{15}$$

其中：$\mathbf{K}$ 是$|U|\times|U|$的核矩阵，$\mathbf{K}(u_q,u_r)$的计算方法为式(11)或者式(13)。

算法 2 所示是谱聚类算法的伪代码。

算法 2　拉普拉斯谱聚类算法

输入：给定 $\mathrm{R}^{|U|\times|U|}$中的 $\mathbf{K}$ 矩阵。

输出：矩阵 $\mathbf{D}$ 与 $\mathbf{L}$ 的评估结果。

1 计算广义特征问题 $\mathbf{L}\Psi=\Lambda\mathbf{D}\Psi$ 的前两个特征向量 Ψ_1 与 Ψ_2，其中 Λ 表示特征值 $\lambda_1,\ldots,\lambda_{|U|}$的对角矩阵。

2 将 Ψ_1 与 Ψ_2 两个特征向量组成矩阵 $\Psi\in\mathrm{R}^{|U|\times2}$。将 Ψ 的各行作为映射空间内的新特征向量。

3 采用 2-means 聚类算法对特征向量进行处理，获得分类的向量。

3.4　恶意用户类的自动识别

大多数恶意用户均会表现出重复且相关的行为，而合法用户则一般表现出不重复且多样化的行为。4.3 节将合法用户与恶意用户进行了分类，下一个任务则是识别恶意用户的攻击类型。

因为恶意用户的消息序列向量的重复性高、多样性低，所以计算类内用户消息序列向量的协方差矩阵。如果协方差矩阵越小，那么该类为恶意用户的可能性越大。算法 3 所示是恶意用户的选择算法。

算法 3　恶意用户类的选择算法

输入：分类后的向量 $\mathbf{C}$。

输出：类 C_1 与 C_2 的类型。

```
1 计算类 C1 与 C2 内投影消息向量的协方差矩阵;
2if (协方差矩阵==0) {
3     该类为恶意用户;
4} else {
5     计算协方差矩阵的特征值;
6     特征值最高的类为恶意用户。
7}
```

算法 4 所示是 DDoS 智能安全系统的总体伪代码。

算法 4　DDoS 智能安全系统的总体伪代码

```
提取 OpenFlow 报文头的信息
按照 2.2 节建立向量模型与矩阵模型
if (时间序列对齐核) {
    设置权重参数 γ 与 ρ,
    计算核矩阵 K，即 ∀(u_q,u_r)∈U×U，可得 K_q,r=K(u_q, u_r)。
    对 K_unnormed 进行对角正则化处理，获得 K。
}
if (用户距离核) {
    计算核矩阵 K，即 ∀(u_q,u_r)∈U×U，可得 K_q,r=K(u_q, u_r)。
}
采用拉普拉斯谱聚类算法(算法 2)将 K 分类，获得结果 C。
采用算法 3 检测 C 的恶意用户。
```

4　实验结果与分析

4.1　实验环境与实验方法

实验环境为 PC 机：Intel Core i7-4770 处理器，3.4 GHz 主频，8 GB 内存。操作系统为 Linux Ubuntu 14.04，交换机系统为 Mininet Version 2.2.26，OpenFlow version 1.3。在实验室搭建了实际的软件定义网络，如图 7 所示。该网络由三个控制器组成，每个控制器包含一个 POX 控制层、64 个主机与 8 个 OpenFlow 交换机。每个 OpenFlow 连接 8 个主机，组成一个子网。POX 是一个快速的轻量级 SDN 控制层，并且支持各种平台，因此实验中采用了 POX 控制层。

采用 MiniEdit 工具建立仿真网络，采用 Open Virtual 交换机(OVS)作为网络交换机，OVS 中已经实现了各种主流的网络接口与网络协议。采用 Scapy 工具产生实际的 flooding 攻击，合法流量中包含随机的数据流。基于 Python 语言编程生成网络流量，使用 Python 的“*randrange*”函数生成随机的源 IP 地址，合法流量的目标 IP 地址范围为 10.0.0.1~10.0.0.64，运行两个 python 脚本，一个负责生成攻击流量，另一个负责生成合法流量。

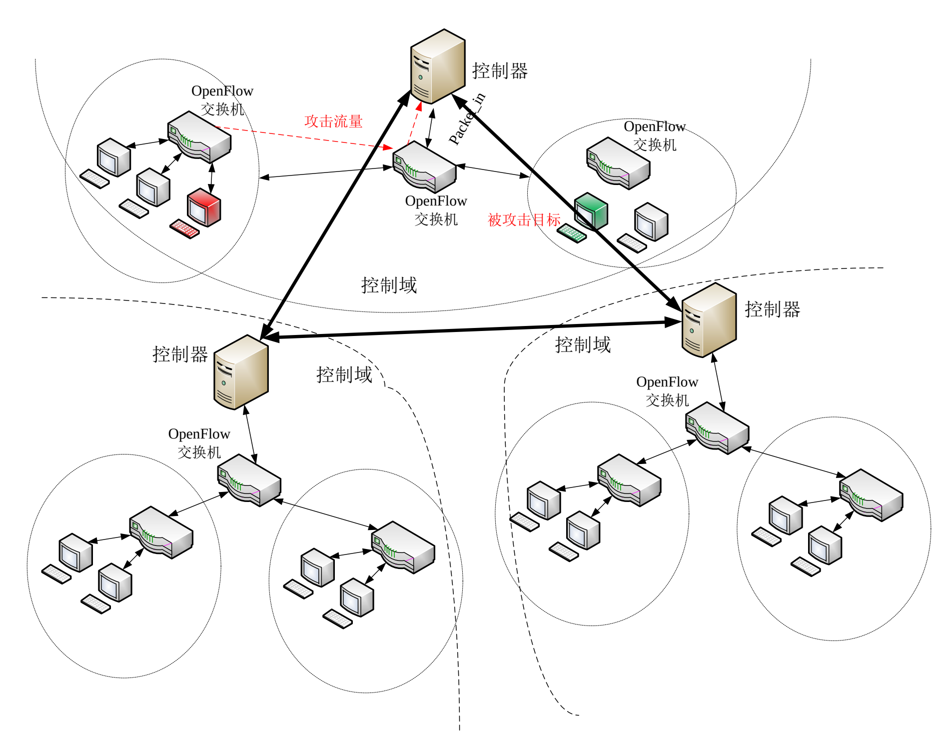


图 7　实验搭建的软件定义网络

Fig. 7　Software defined network constructed in the experiment

4.1.1 DDoS 安全系统的性能评估指标

采样精度、召回率与 F-score 三个指标评估 DDoS 安全系统的性能，分别定义为

精度 ＝ 正确检测的攻击/所有检测的攻击　(16)

召回率 ＝正确检测的攻击/ 样本的总数量　(17)

F-score ＝精度*召回率*2/(精度+召回率)　(18)

4.1.2 测试例

为了测试本系统对 DDoS 攻击安全性，考虑了两个实验测试例。第一个测试例为单目标攻击实验：在一个主机上运行三种不同流量比例的 DDoS 攻击，三种攻击流量比例分别为 10%、20%、30%；第二个测试例为子网目标攻击的实验：在一个子网中运行三种不同比例的 DDoS 攻击，三种攻击流量比例分别为 10%、20%、30%。

4.1.3 仿真参数设置

表 1 所示是实验中的参数设置。因为实验中共有 64 个主机，所以本系统的窗口大小设为 80。根据预处理实验的结果，本实验环境下 k=5、λ=1、β=4、c=1 获得了较好的检测性能。

表 1　测试例的参数设置

Table 1　Parameters of test cases

参数	取值
报文类型	UDP
窗口大小	80
目标端口	80
源端口	2
合法流量间隔	0.1 秒
攻击流量间隔	0.025 秒
报文总量	6500
数据内容	随机生成
k	5
λ	1
β	4
c	1

4.2　实验结果与性能分析

4.2.1 单目标攻击实验的结果

因为本系统采样周期对检测精度存在一定的影响，所以本文考虑了三个采样周期，分别为 1 s、3 s、5 s。将本算法与 KNNDD[6]、VNDD[7]两个同类型 DDoS 检测系统进行横向比较，综合地评估本系统的性能。每组实验独立地运行 30 次，将 30 次实验的结果作为最终的统计结果。

图 8 所示是三个安全系统的单目标攻击实验的结果。从图中可看出，本系统 1 s 采样周期的精度、召回率与 F-score 均低于 KNNDD、VNDD 两个系统，而本系统 3 s 与 5 s 采样周期的性能则具有明显的优势。

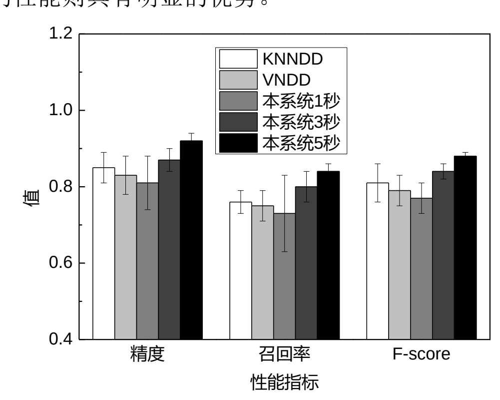


图 8　三个安全系统的单目标攻击实验的结果

Fig. 8　Result of single target attack of three security systems

4.2.2 子网目标攻击实验的结果

本文考虑了三个采样周期，分别为 1 s、3 s、5 s。将本算法与 KNNDD[6]、VNDD[7]两个同类型 DDoS 检测系统进行横向比较，综合地评估本系统的性能。每组实验独立地运行 30 次，将 30 次实验的结果作为最终的统计结果。

图 9 所示是三个安全系统的子网目标攻击实验的结果。从图中可看出，本系统 1 s 采样周期的精度、召回率与 F-score 均低于 KNNDD、VNDD 两个系统，本系统 3 s 采样周期的性能与 KNNDD 算法接近，而系统 5s 采样周期的性能表现

出明显的优势。

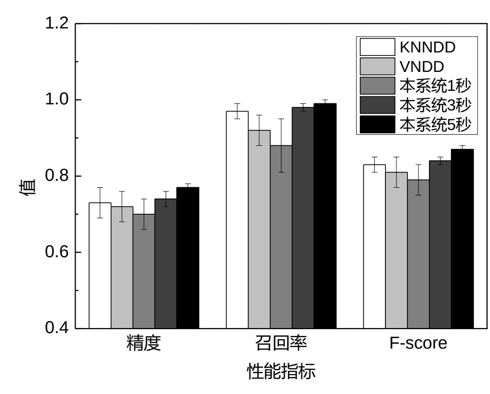


图 9 三个安全系统的子网目标攻击实验的结果

Fig. 9 Result of sub-network target attack of three security systems

4.2.3 安全系统的计算效率

根据算法 4 的处理流程，本系统主要有三个模块组成，分别为计算距离核、计算时间序列对齐核、谱聚类处理。实验中分别统计了三个模块的平均处理时间，如图 10 所示。从图中可看出，三个不同的采样周期下，本系统的处理时间较为稳定，总时间约为 3.5 s，满足实时检测的要求。

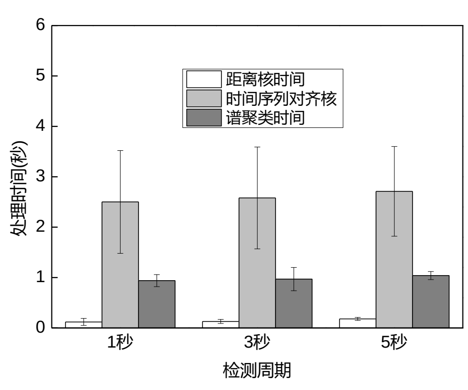


图 10 三个模块的平均处理时间

Fig. 10 Average processing time of three models

5 结束语

DDoS 攻击的出现往往伴随着流量模式的剧烈变化，本系统将消息流特征的明显变化作为一个潜在的 DDoS 攻击。本系统能够检测出 DDoS 攻击行为并识别出攻击者。本系统属于无监督方法，利用观察的消息流量类型与数据量，并不需要额外的信息。基于真实软件定义网络进行了实验，结果显示该安全系统实现了较高的检测准确率，并且实现了理想的处理时间。未来将考虑在实际大规模软件定义网络中进行实验，评估本系统的有效性与性能。

参考文献：

[1] Saied A, Overill R E, Radzik T. Detection of known and unknown DDoS attacks using artificial neural networks [J]. Neurocomputing, 2016, 172 (C): 385-393.

[2] Kolias C, Kambourakis G, Stavrou A, *et al*. DDoS in the IoT: mirai and other botnets [J]. Computer, 2017, 50 (7): 80-84.

[3] 王蒙蒙，刘建伟，陈杰，等. 软件定义网络：安全模型、机制及研究进展 [J]. 软件学报, 2016, 27 (4): 969-992. (Wang Mengmeng, Liu Jianwei, Chen Jie, *et al*. Software defined networking: security model, threats and mechanism [J]. Journal of Software, 2016, 27 (4): 969-992.)

[4] 王秀磊，陈鸣，邢长友，等. 一种防御 DDoS 攻击的软件定义安全网络机制 [J]. 软件学报, 2016, 27 (12): 3104-3119. (Wang Xiulei, Chen Ming, Xing Changyou, *et al.* Software defined security networking mechanism against DDoS attacks [J]. Journal of Software, 2016, 27 (12): 3104-3119.)

[5] 姚琳元，董平，张宏科. 基于对象特征的软件定义网络分布式拒绝服务攻击检测方法 [J]. 电子与信息学报, 2017, 39 (2): 381-388. (Yao Linyuan, Dong Ping, Zhang Hongke. Distributed denial of service attack detection based on object character in software defined network [J]. Journal of Electronics & Information Technology, 2017, 39 (2): 381-388.)

[6] 肖甫，马俊青，黄洵松，等. SDN 环境下基于 KNN 的 DDoS 攻击检测方法 [J]. 南京邮电大学学报 ：自然科学版 , 2015, 35 (1): 84-88. (Xiao Fu, Ma Junqing, Huang Xunsong, *et al.* DDoS attack detection based on KNN in software defined networks [J]. Journal of Nanjing University of Posts and Telecommunications ： Natural Science , 2015, 35 (1): 84-88.)

[7] Dong P, Du X, Zhang H, *et al.* A detection method for a novel DDoS attack against SDN controllers by vast new low-traffic flows [C]// Proc of IEEE International Conference on Communications. 2016: 1-6. VNDD

[8] Kalkan K, Gur G, Alagoz F. Defense mechanisms against DDoS attacks in SDN environment [J]. IEEE Communications Magazine, 2017, 55 (9): 175-179.

[9] 杨君刚，王新桐，刘故箐. 基于流量和 IP 熵特性的 DDoS 攻击检测方法 [J]. 计算机应用研究, 2016, 33 (4): 1145-1149. (Yang Jungang, Wang Xintong, Liu Guqing. DDoS attack detection method based on network traffic and IP entropy [J]. Application Research of Computers, 2016, 33 (4): 1145-1149.)

[10] Šulák V, Helebrandt P, Kotuliak I. Performance analysis of OpenFlow forwarders based on routing granularity in OpenFlow 1. 0 and 1. 3 [C]// Proc of Open Innovations Association. 2017: 236-241.

[11] Davis J V, Kulis B, Jain P, *et al.* Information-theoretic metric learning [C]// Proc of International Conference on Machine Learning. 2007: 209-216.

[12] Cuturi M, Vert J, Birkenes O, *et al.* A kernel for time series based on global alignments [C]// Proc of IEEE International Conference on Acoustics, Speech and Signal Processing. 2006: II-413-II-416.

[13] Luxburg U V. A tutorial on spectral clustering [J]. Statistics & Computing, 2007, 17 (4): 395-416.